

WE CLAIM

1. A method for designing an initiator in an integrated circuit, said initiator being connected to an interconnect and arranged to issue requests, said method comprising the steps of:

defining if the initiator or the interconnect is to be responsible for ordering responses to requests issued by said initiator;

defining the maximum number of requests which are permitted to be outstanding at the same time; and

defining if a delay stage is required in said initiator port.

2. A method as claimed in claim 1, wherein said number of requests which are permitted to be outstanding are defined if the interconnect is responsible for ordering.

3. A method for designing a target in an integrated circuit, said target being connected to an interconnect and arranged to generate responses to requests, the method comprising the steps of:

defining if the target or the interconnect is responsible for ordering responses;

defining the maximum number of possible outstanding requests which can be supported by said target; and

defining if a delay stage is required in said target port.

4. A method as claimed in claim 3, wherein said the step of defining the maximum number of possible outstanding requests is performed only if the interconnect is responsible for ordering the responses.

5. A method for designing an interconnect having routing resources, said interconnect arranged to allow initiators to send requests to targets, said method comprising the steps of defining:

the number of routing resources between the initiator and the target;

the arbitration method for arbitrating between requests; and
the association between the routing resources and the targets.

6. A method as claimed in claim 5, wherein said method further comprises the step of determining if a delay is required after arbitration.

7. A method for designing an interconnect having routing resources, said interconnect arranged to allow targets to send responses to initiators in response to requests from initiators, said method comprising the steps of defining:

the number of routing resources between the target and the initiator;

the arbitration method for arbitration between responses; and

the association between the routing resources and the initiator.

8. A method as claimed in claim 7, wherein said method further comprises the step of determining if a delay is required after arbitration.

9. A method of designing an arbiter in an integrated circuit comprising initiators, targets and an interconnect, said arbiter being provided between said initiators and said interconnect, said method comprising the steps of:

using an arbitration model having a plurality of different arbitration methods and selecting one of the plurality of arbitration methods available in said model.

10. A method as claimed in claim 9, wherein the method further comprises selecting if a delay is to be provided after arbitration has been performed.

11. A method of designing an arbiter in an integrated circuit comprising initiators, targets and an interconnect, said arbiter being provided between said targets and said interconnect, said method comprising the steps of:

using an arbitration model having a plurality of different arbitration methods and selecting one of the plurality of arbitration methods available in said model.

12. A method as claimed in claim, wherein the method further comprises selecting if a delay is to be provided after arbitration has been performed.

13. A model of a initiator to be used in designing an integrated circuit in which an initiator is arranged to send requests to one or more targets via an interconnect, said model comprising:
an address decode stage for identifying the target for which a given message is intended; and

a dependency stage for determining the allowability of a request, the operation of said dependency stage being selectable, said dependency stage being such that the model supports an arrangement where the initiator or the interconnect is responsible for maintaining the order of responses from a target to the requests.

14. A model as claimed in claim 13, wherein a retime stage is provided in said model, the retime stage arranged to provide a delay or no delay.

15. A model as claimed in claim 13, wherein in an access queue is provided for storing requests for which responses have not been received.

16. A model as claimed in claim 15, wherein the maximum number of requests which can be stored in the queue is definable.

17. A model of a target to be used in designing an integrated circuit in which one or more initiators are arranged to send requests to a target and the target is arranged to send responses to the requests via an interconnect, said model comprising:

a locking stage which permits locked transactions to occur if required; and

a decode state which decodes information stored in a queue into an address for the response.

18. A model as claimed in claim 17, wherein said model comprises an access queue which store information on the requests received by the target.

19. A model as claimed in claim 18, wherein said the maximum number of outstanding request which can be stored in said queue is definable.

20. A model as claimed in claim 17, wherein said queue is in the initiator.

21. An arbiter comprising:

an input for receiving a plurality of requests from a plurality of sources;

arbitration logic for arbitrating between said requests in accordance with an arbitration method;

a store for storing information defining the priority of said sources,

whereby said arbitration logic is arranged to update said store after arbitration to define the priority of said sources for a subsequent arbitration.

22. An arbiter as claimed in claim 21, wherein the store comprises a table in which the position of the source in the table determines the priority of the source.

23. An arbiter as claimed in claim 21, comprising packet control logic for maintaining the ordering of packets containing the requests.

24. A queue for storing a predetermined maximum number of outstanding requests, said queue comprising:

a store for storing the outstanding requests, a location being provided for each of the maximum number of outstanding requests;

logic for determining if each of said locations contains an outstanding request and if so to provide a signal indicating that the queue is full.

25. A method for designing an initiator in an integrated circuit, said initiator being connected to an interconnect and arranged to issue requests, said method comprising the steps of:

defining if the initiator or the interconnect is to be
responsible for ordering responses to requests issued by said
initiator; and

defining if a delay stage is required in said initiator port.

2024.4.14.10.4004.05